# Chapter 6

# Bifurcation analysis of driven cavity flow by the Newton-Picard method

## 6.1 Introduction

In this chapter we will study the transition from steady to (quasi) periodic flow. The flows under consideration are governed by the incompressible Navier-Stokes equations. Such studies are not only interesting from a theoretical point of view, but from a practical one as well. For example, consider a flow around an object which becomes periodic at a certain Reynolds number. At that stage the flow will exert a periodic force on the object, which may give such strong vibrations that, due to fatigue of the material, the object will eventually get damaged. In such a case one could change the form of the object in order to improve its aerodynamic properties such that the periodic behaviour is delayed to a much higher Reynolds number which does not or only rarely occur in practice.

The study on the transition will be done for the lid-driven cavity. This problem is a well known test case for numerical methods for solving the Navier-Stokes equations. Nevertheless, little is known about its bifurcation behaviour. Cazemier et al. [12] have performed a bifurcation analysis for the lid-driven cavity using a low-dimensional dynamical system resulting from a Galerkin projection of the Navier-Stokes equations on a basis obtained by a proper orthogonal decomposition (or singular value decomposition) of snapshots of the flow on a very fine grid (see Section 6.2.2). The low-dimensional system admits to compute the monodromy matrix, the order of which is equal to the number of degrees of freedom in the system, whose eigenvalues determine the bifurcation behaviour. They observed that the behaviour of the low-dimensional dynamical system predicted the behaviour of the high-dimensional system on the very fine grid reasonably well.

For our bifurcation analysis we have used the software tool PDECONT, which has been developed by Lust and Roose [32]. This tool is based on the Newton-Picard method which in itself is a generalization of the RPM of Shroff and Keller [42]. In this method the monodromy matrix restricted to the space consisting of the most critical modes is constructed. Therefore, the tool allows to study the bifurcation behaviour of the high-dimensional system itself. In the tool the user has to specify the right-hand side and the flow, i.e. a map which gives the time-dependent solution of the PDE for a given time, Reynolds number and initial solution. We have computed this flow by using an implicit

time-integration method where MRILU (Chapter 4 and [8]) has been applied to solve the occurring linearized systems. Due to limited computer resources we have used a somewhat coarser grid than Cazemier et al. Nevertheless, the results are very similar.

## 6.2 Problem description and results of others

In this section we will describe the problem and summarize the information about its bifurcation behaviour found by others.

### 6.2.1 Lid-driven cavity

The problem under consideration is the lid-driven cavity problem; the flow in a square cavity with constant moving lid. The domain and boundary conditions of this problem are shown in Figure 6.1.

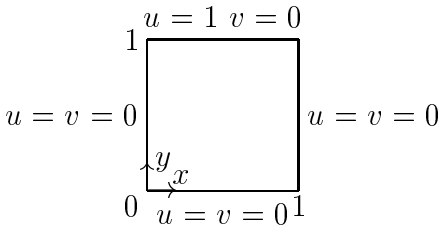


Figure 6.1: Geometry for the lid-driven cavity problem.

The flow is described by the two-dimensional incompressible Navier-Stokes equations, which in conservation form are given by

$$\begin{aligned}
&\int_\Gamma \vec{u}.n \; \mathrm{d}\Gamma = 0, \\
&\int_\Omega \frac{\partial u}{\partial t} \; \mathrm{d}\Omega + \int_\Gamma (u\vec{u} - \nu\nabla u).n \; \mathrm{d}\Gamma = -\int_\Gamma p(n.e_1) \; \mathrm{d}\Gamma, \\
&\int_\Omega \frac{\partial v}{\partial t} \; \mathrm{d}\Omega + \int_\Gamma (v\vec{u} - \nu\nabla v).n \; \mathrm{d}\Gamma = -\int_\Gamma p(n.e_2) \; \mathrm{d}\Gamma,
\end{aligned}$$

for arbitrary domain $\Omega$ with boundary $\Gamma$.

### 6.2.2 Results by Cazemier

The flow in a lid-driven cavity was also studied by Cazemier et al. [11, 12]. In their work a direct numerical simulation is performed at Reynolds number 22,000 on a $250 \times 250$ grid (about 200,000 unknowns) with a fourth-order finite-volume discretization. This Reynolds number is far beyond the transition point from steady to periodic flow; the flow is already chaotic. From this simulation 700 snapshots are taken at constant time intervals (5 sec). By a proper orthogonal decomposition (POD), or in linear algebra terms a singular value

| $f$ | 0.27 | 0.44 | 0.11 | 0.16 | 0.32 | 0.53 | 0.70 | 0.06 | 0.38 | 0.60 | 0.88 |
|---|---|---|---|---|---|---|---|---|---|---|---|

Table 6.1: Occurring frequencies (Hz) in DNS calculation at Re=10,000.

decomposition, of these snapshots an orthogonal basis is constructed. They showed that a basis of dimension 80 is enough to capture 95% of the energy in the system. Then, these 80 basis functions are used for a Galerkin approximation of the Navier-Stokes equations. The thus derived low-dimensional dynamical system is used to study the bifurcation behaviour. At certain Reynolds numbers the results of this dynamical system were verified by performing a DNS. The results were qualitatively similar, but quantitatively different.

The DNS results in [11] yield some important data to compare our results with. First, the transition from a stationary to a periodic flow is observed at Re=7972, the emerging period $T$ is 2.2 seconds. Secondly, at Re=10,000 a plot is given of the power spectral density of the time signal. We clearly see a frequency 0.44 Hz (about the reciprocal of the period 2.2) in the plot. In Table 6.1 the most important frequencies, as good as we can read it from the plot, in that signal are listed in order of significance. The signal contains already a lot of frequencies, among which there will be combination frequencies. In our experiments (see Section 6.5) we will see that for this Reynolds number the steady state solution has already 5 unstable modes. Many of the occurring frequencies shown above can be found as combination frequencies of these modes.

The analysis with the low-dimensional dynamical system showed at Re=7819 a transition from a stationary to a periodic flow with time period 1.63 seconds, which corresponds to a frequency of about 0.60 Hz. This frequency is quite different from the frequency 0.44 Hz obtained from the DNS, which is explained in Cazemier's thesis [11] from the fact that the POD basis has been obtained at Re=22,000. However, our experiments also indicate that the frequency 0.60 Hz is rather important.

### 6.2.3 Results by Poliashenko and Aidun

Poliashenko and Aidun described in [38] a direct method to compute a simple bifurcation. They solve an augmented system which give the critical parameter value at once, avoiding a continuation process. Similar techniques are discussed in [1]. This direct method was applied to compute the first bifurcation of the flow in the lid-driven cavity. Based on the results on some grids they concluded that this critical value is at $7763 \pm 2\%$. By using Richardson extrapolation on the results of their finest two grids we come to the value 7952 which is quite close to the value 7972 found by Cazemier et al. It surprised us that on a $57 \times 57$ grid using biquadratic finite elements such an accurate determination of the bifurcation point is possible.

Another interesting result they obtained is that for a square cavity the first (Hopf) bifurcation is supercritical, which means that the solution is unique before the first bifurcation. They show a result of a non-square cavity with height-width ratio 1.5 where this is not the case; next to the steady state there are a stable and an unstable periodic solution.

## 6.3 Discretization

In general a parameter dependent autonomous partial differential equation (PDE) is considered. With the method of lines this PDE is first discretized in space, resulting in a large system of ODEs

$$\begin{aligned} &\frac{dx}{dt} = f(x,\gamma) \quad t > 0, \\ &x(0) = x_0, \end{aligned}$$

with $x \in R^n$ and $\gamma$ the parameter. To solve this system and perform a bifurcation analysis we use the software tool PDECONT. This tool requires from the user two subroutines: one which computes the right-hand side $f(x,\gamma)$ and one which computes the flow $\varphi(x_0,t,\gamma)$, i.e. the solution of the PDE at time $t$ for a given initial solution $x_0$. To compute this flow we employ as time integration the $\theta$-method. In the following we treat, according to the method of lines, the discretization of the Navier-Stokes equations in space and time separately.

### 6.3.1 Space-discretization

In order to perform an accurate bifurcation analysis, we wish to preserve the stability of the Navier-Stokes equations when discretizing the system. In order to obtain this, the convective terms have to be discretized skew-symmetric [53]. In the continuous case convection corresponds to a skew-symmetric and diffusion to a symmetric positive definite operator. When preserving these properties in the discretized case, the resulting coefficient matrix will be positive definite. In [53] it is shown that if after semi-discretization the coefficient matrix is positive definite the energy of the solution is decreasing in time. This implies that the semi-discretized system is stable. We will use the symmetry-preserving finite-volume discretization as described in Section 4.3.1.

The starting point in this discretization is a central discretization of the convective terms, but by adding artificial diffusion to the real diffusion an upwind discretization can be obtained. In our computations we want to eliminate the influence of artificial diffusion on the results. Hence, we set it equal to zero. We will use a $128 \times 128$ grid, and refine the grid near the walls in order to prevent unphysical wiggles. This refinement is determined by the mapping

$$y_s = \frac{1}{2}(1 + \tanh(2s(y_u - \frac{1}{2}))/\tanh(s)).$$

This expression maps grid points $y_u$ from a uniform grid to those of a stretched grid $y_s$. As stretching factor we use $s = 1.5$ in both $x$- and $y$-direction.

### 6.3.2 Time-discretization

For the time-discretization we used the $\theta$-method, which is given by

$$x^{n+1} = x^n + \Delta t f((1-\theta)x^n + \theta x^{n+1}),$$

with $\theta \in [0,1]$. For $\theta = 1/2$ and $\theta = 1$ this is the implicit midpoint and backward Euler method, respectively. The choice of $\theta$ in the application is delicate and will be considered separately below. As initial guess for the solution at time step $n+1$ we used the extrapolation formula $x^{n+1} = 2x^n - x^{n-1}$. The implicit relations are solved by the Newton method and the thereby occurring linear equations by MRILU. In the Newton-Picard method we are computing perturbations of a steady state over a certain time period. It appeared that those perturbations are so small that one MRILU factorization for the whole time period sufficed.

**Choice of $\theta$**

In the time-continuous case the high-frequency components are highly damped, and components with eigenvalues with a large imaginary part and a small positive real part are unstable. When using a time discretization, for instance the $\theta$-method, we wish to preserve these properties. We will consider the influence of the choice of $\theta$ on the damping of the different components.

Suppose we want to study the effect of perturbations of a steady state over a time period $T$ with $n$ equal time steps, hence $\Delta t = T/n$. Then, for each eigenvalue $\lambda$ of the Jacobian matrix of $f$ we have the following amplification factor $r$

$$r = [(1 + (1-\theta)\lambda\Delta t)/(1 - \theta\lambda\Delta t)]^n. \tag{6.1}$$

The Newton-Picard method, discussed below, computes the most dominant $r$'s, i.e. those $r$'s which are in magnitude larger than a certain user specified value. We want that the corresponding dominant subspace contains the same perturbations as those dominating in the time-continuous case. So on the one hand we want $\theta = 1/2$, since then the important eigenvalues near the imaginary axis are not damped too strongly, avoiding that they will not be treated as dominant by the Newton-Picard method and consequently will become invisible. On the other hand we want $\theta$ larger than $1/2$ in order to damp the $\lambda$'s large in magnitude. So we try to find $\theta$ close to $1/2$ such that for $\lambda$'s large in magnitude the magnitudes of the corresponding $r$'s are less than those corresponding to $\lambda$'s close to the imaginary axes. For large $\lambda\Delta t$ we have from (6.1) the approximation

$$|r| \approx [(1-\theta)/\theta]^n \approx \exp[-4(\theta - 1/2)n]. \tag{6.2}$$

We will clarify the above with a numerical example. Assume we wish to compute all $r$'s which are in magnitude larger than 0.95. In our computations the number of time steps we used is about 10 to 20. For 10 time steps and $\theta = 0.51$ we find for $\lambda$'s large in magnitude $|r| \approx 0.7$, which is only slightly below 0.95 and therefore a further decrease of $\theta$ is not desirable, since this would increase $r$. For eigenvalues with imaginary part of about 2 to 4 and close to the imaginary axis we observed in our computations an influence of the time discretization on the damping when $\theta = 0.51$. Hence, on this ground we would like to decrease $\theta$. We see that these considerations for the choice of $\theta$ are in conflict. When the time step is halved $n$ is doubled and we are allowed (see (6.2)) to decrease $\theta$ to 0.505 in order to keep the same damping behaviour for the high-frequency components.

**Computation of eigenvalues from $r$**

At this place we discuss an application of equation (6.1) which can be used to compute where the stationary solutions of the semi-discretized equations become unstable. As mentioned before the Newton-Picard method will compute the value of $r$. From equation (6.1) we can compute the corresponding eigenvalue $\lambda$. Due to the $n$-th root that has to be taken $\lambda$ is multi-valued. Only one of these values is the sought eigenvalue. This true eigenvalue is of course unchanged under perturbation of the time period $T$. The others values do change under such a perturbation. To see this, consider the expression

$$r = \exp(\lambda T),$$

which is the continuous variant of equation (6.1) which we use here to simplify the analysis. Now, for the true eigenvalue $\hat{\lambda}$ we have

$$\frac{dr}{dT} = \hat{\lambda}\exp(\hat{\lambda}T), \tag{6.3}$$

which we will observe after a perturbation of $T$. The pseudo eigenvalues mathematically yield after a perturbation of $T$

$$\frac{dr}{dT} = T\exp(\lambda T)\frac{d\lambda}{dT} + \lambda\exp(\lambda T).$$

When we equate this to the observed one (6.3) we find

$$\frac{d\lambda}{dT} = (\hat{\lambda}\exp[(\hat{\lambda}-\lambda)T] - \lambda)/T.$$

So in general $\frac{d\lambda}{dT}$ differs from zero if $\lambda$ differs from $\hat{\lambda}$ and hence we can detect $\hat{\lambda}$ by changing $T$ a little for a fixed parameter (the Reynolds number). The most constant value among the solutions is the sought eigenvalue.

This can be used during the continuation process as well. For the steady state we see that the Newton-Picard algorithm also changes $T$. By picking the most constant value in the set of computed values for $\lambda$ we can identify the eigenvalue.

## 6.4 PDECONT

In this section the so-called Newton-Picard method which is at the basis of PDECONT will be described in short (see [32] for an extensive treatment). The underlying idea of the method can already be found in [27, 42], where it is applied to steady computations. Shroff and Keller called their method the recursive projection method (RPM).

To introduce the underlying idea we confine ourselves to the steady case. If we use a time-integration method, eventually the error is dominated by a set of slowly decaying (periodic) components. It is possible to make a basis for this set of slowly decaying components which forms an invariant subspace of limited dimension. The update in this invariant subspace and its orthogonal complement will be computed separately. In the former Newton's method is used and in the latter the time integration is still employed.

To explain the Newton-Picard method we begin with a description of the RPM, in order to point out the main ideas.

### 6.4.1 RPM

Suppose we have a fixed-point problem

$$x = g(x).$$

One may think here of a steady-state computation by means of a time-integration method over some period, or of detecting a periodic solution of a PDE. The fixed-point iteration $x_{n+1} = g(x_n)$ may converge slowly or even diverge when the eigenvalues of the Jacobian of $g$ are in magnitude close to 1 or larger than 1.

To stabilize this procedure the space will be split in two orthogonal subspaces $\mathcal{P}$ and $\mathcal{Q}$ with projectors $P$ and $Q = I - P$. The subspace $\mathcal{P}$ is of low dimension, and corresponds to the slowly converging and unstable modes. Let $p_0 = Px_0$ and $q_0 = Qx_0$, then we iterate according to

$$\begin{aligned} p_{i+1} &= p_i + C_1(Pg(x_i) - p_i), \\ q_{i+1} &= q_i + C_2(Qg(x_i) - q_i), \\ x_{i+1} &= p_{i+1} + q_{i+1}. \end{aligned}$$

Here, $C_1$ and $C_2$ should be chosen such that good convergence is obtained in both subspaces. In the paper of Shroff and Keller [42] $C_2$ is the identity matrix, yielding a Picard iteration in $\mathcal{Q}$, and $C_1$ is minus the inverse of the Jacobian of $Pg(x_i) - p_i$,

$$-(\frac{\partial(Pg(p_i + q_i) - p_i)}{\partial p_i})^{-1},$$

resulting in the Newton method in $\mathcal{P}$.

In the iteration we need the projectors $P$ and $Q$. Let the columns of $V$ be the orthogonal vectors spanning $\mathcal{P}$. Then the projectors are given by $P = VV^T$ and $Q = I - P = I - VV^T$. In the actual computations we do not work with $p$ but with its representation $\hat{p}$ in the basis $V$, every $p \in \mathcal{P}$ can be written as $p = V\hat{p}$. Hence, we can rewrite the $p$ update equation as

$$V\hat{p}_{i+1} = V\hat{p}_i + C_1(VV^T g(V\hat{p}_i + q_i) - V\hat{p}_i).$$

Premultiplying this equation by $V^T$ yields

$$\hat{p}_{i+1} = \hat{p}_i + V^T C_1 V(V^T g(V\hat{p}_i + q_i) - \hat{p}_i).$$

In order to obtain a good convergence in the subspace $\mathcal{P}$ we zero the Jacobian of the right-hand side in this formulation. This gives

$$V^T C_1 V = (I - V^T g_x V)^{-1}.$$

Here, the directional derivatives $g_x V_i$ where $V_i$ is the $i$-th column of $V$ can be found by numerical differentiation

$$g_x(x_n)V_i = (g(x_n + \varepsilon V_i) - g(x_n))/\varepsilon. \tag{6.4}$$

*Remark* In the periodic case the matrix $V^T g_x V$ is an approximation to the monodromy matrix restricted to $\mathcal{P}$.

We are left with the problem of finding $V$. Its columns build the invariant subspace $\mathcal{P}$ which usually is the space corresponding to the dominant eigenvalues of $g_x$. $V$ can be found in various ways. We will return to this problem in the next section.

Note that the presented iteration is in general more contractive than the original fixed-point problem. Even if the fixed-point problem is mildly unstable, i.e. only a few eigenvalues of $g_x$ are in magnitude larger than one, then the RPM is converging if the eigenvectors corresponding to those unstable eigenvalues are in $\mathcal{P}$.

### 6.4.2 Newton-Picard method

Lust and Roose reconsidered and generalized the RPM for periodic problems starting from a linear algebra point of view and called their method the Newton-Picard method. We will describe the part of the algorithm that we have used in our computations. For the other features we refer to [32, 31]. An outline of the algorithm including a continuation process is given in Algorithm 6.1. In the remainder of this section we will clarify the main part of this algorithm.

Given an initial guess for a solution at a given parameter

1. Compute the solution accurately
   (a) Construct an invariant subspace
   (b) Compute the correction
   (c) Check convergence
   (d) If not converged go to (a)
2. Increment the parameter
3. Generate a guess for the solution at the new parameter value
4. Go to 1 or stop if goal is reached

Algorithm 6.1: Newton-Picard method.

#### Construction of the invariant subspace

The invariant subspace is found by a variant of orthogonal iteration with Ritz acceleration (see [21, page 423]) or subspace iteration with projection [39]. The iteration process is given by Algorithm 6.2. Here, $M$ stands for the monodromy matrix or in terms of the previous section $g_x$. If Steps 2 to 4 are omitted and $V = W$, we just have the standard orthogonal iteration process (a generalization of the Power method). The effect of these

Given some $V_0$ with $V_0^T V_0 = I$ and $M$
Set $k = 1$ and perform the following steps

1. Apply $M$ to $V_{k-1}$: $W_k = MV_{k-1}$
2. Compute the restriction of $M$ to the space spanned by $V_{k-1}$: $M_k = V_{k-1}^T W_k (\equiv V_{k-1}^T M V_{k-1})$
3. Convert $M_k$ to its real Schur normal form where the eigenvalues on the diagonal of R are in decreasing order of magnitude: $U^T M_k U = R$
4. Combine columns in $W_k$ such that they correspond to the order of the eigenvalues: $V_k = W_k U$
5. Orthogonalize $V_k$
6. Increment $k$ and go to 1 or stop if converged

Algorithm 6.2: Construction of invariant subspace.

steps is the following. They construct a basis for the subspace such that the eigenvector (eigenspace) corresponding to the eigenvalue (eigenvalue pair) largest in magnitude is on the first position, repeating itself for the remaining eigenvalues. This is an acceleration of the orthogonal iteration where eventually also the eigenvectors will appear in the desired order.

In the standard acceleration [21, page 423] Step 2 is replaced by first orthogonalizing $W$, which allows to omit Step 5, and then computing $M_k = \tilde{W}_k^T M \tilde{W}_k$. This requires a new application of $M$, which will slow down the construction by almost a factor two. The difference is that in the algorithm used the reordering is based on the old $V$, this difference has only a limited effect on the convergence. Eventually, when the respective subspaces are converging, $U$ tends to the identity matrix, and the eigenvalues on the diagonal of $R$, i.e. the eigenvalues of the monodromy matrix $M$ restricted to the invariant subspace, are the amplification factors mentioned in subsection 6.3.2.

*Convergence behaviour* In exact arithmetic we could, by omitting the orthogonalization, simplify the construction to the iteration $V_k = MV_{k-1}$, requiring only that the distance [21, page 76] of $V_0$ and the dominant eigenspace is less than one. If $d$ is the dimension of the space spanned by $V_k$ then the $i$-th Ritz value converges as $(\lambda_{d+1}/\lambda_i)^k$. Here the Ritz value can be obtained by orthogonalizing $V_k$, restricting $M$ to that space as in Step 2, and computing the eigenvalues of that restriction in decreasing order (the $i$-th eigenvalue of this matrix is the $i$-th Ritz value).

In order to get the same result in the presence of rounding errors, $V_k$ has to be orthogonalized now and then (usually at each step), resulting in orthogonal iteration. In orthogonal iteration the eigenvectors will converge in order of decreasing magnitude of the

eigenvalues. In order to accelerate the convergence of the eigenvectors, one recombines the columns in $V_k$ when the Ritz values are computed (here at every step), yielding the same convergence in the eigenvectors as in the Ritz values. For more considerations we refer to [39, Ch.5].

*Parallelization* The computation of $MV$ is well parallelizable. Since this is the most time consuming operation of the code this is very attractive. $V$ consists of the order of 30 columns. So almost a factor 30 can be gained, since the work is strongly balanced.

*Stopping criterion and locking* For the stopping criterion the matrix

$$(I - VV^T)MV$$

is considered. Here $(I - VV^T)$ is the projector on the orthogonal complement of $V$. Hence, if $V$ is an invariant subspace then this expression is zero. It is not necessary that all columns of $V$ have converged. The user specifies a lower bound for the magnitude of the eigenvalues that must converge and a tolerance which is used as follows. If the eigenvalues corresponding to the space spanned by the first $k$ columns of $V$ are larger than the lower bound and if the norm of the first $k$ columns of $(I - VV^T)MV$ is less than the tolerance then these $k$ columns are assumed to have converged. Of course, the space $V$ must be taken large enough such that all wanted eigenvectors are in it. A larger space than strictly necessary is even beneficial as we have seen in the paragraph on the convergence behaviour.

As we have seen the eigenvectors converge in order, this can be exploited by freezing (or locking) a converged vector. In this case Step 1 of Algorithm 6.2 is only performed for the not-converged part. The other steps are still applied to all vectors in $V$.

*Remark* We did not make use of the ability of the program to add and delete vectors in $V$. In the case of adding, random vectors are employed. We observe an algebraic effect in the convergence of such a vector which delays the overall convergence considerably. In our view first a number of iterations on new (random) vectors have to be done while the old ones are locked. In the course of our study we got sufficient information on how to choose the dimension of the space spanned by the columns of $V$.

### Computation of the correction

In order to describe the computation of the correction we rewrite our problem. Let us describe the solution of the system in terms of the flow $\varphi$: $x(t) = \varphi(x(0), t, \gamma)$. So the solution depends on the initial solution and the parameter $\gamma$. If a periodic solution exists then the flow must have a fixed point

$$x = \varphi(x, T, \gamma). \tag{6.5}$$

Note that the fixed point is not unique, if $x$ is a fixed point then $\varphi(x, t, \gamma)$ is one as well. The solution is often made unique by imposing a so-called phase condition, which we write formally as

$$s(x, T, \gamma) = 0. \tag{6.6}$$

*Remark* By substituting $\varphi(x,t,\gamma)$ in the fixed-point equation and taking the derivative with respect to $t$ we see immediately that $\varphi_t(x,t,\gamma)$ is an eigenvector of $\varphi_x(\varphi(x,t,\gamma),T,\gamma)$ with eigenvalue 1. For $t=0$, this means we have the eigenvector $\varphi_t(x,0,\gamma)$ and the matrix $\varphi_x(x,T,\gamma)$. In the latter case the eigenvector is equal to $f(x,\gamma)$. We will use this result later on.

Of course, one likes to solve the above two coupled equations (6.5) and (6.6) with the Newton method. The Newton-correction equation is given by

$$\begin{bmatrix} \varphi_x - I & \varphi_T \\ s_x & s_T \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta T \end{bmatrix} = - \begin{bmatrix} r \\ s \end{bmatrix},$$

where $r$ is the residual of the fixed-point equation. Here, $\varphi_x$ is an approximation to the so-called monodromy matrix, therefore it is usually denoted by $M$. As explained in the remark above, $\varphi_T$ can be approximated by $f(x,\gamma)$. The matrix $\varphi_x$ is full, and hence, for large systems, the Newton method will be too expensive.

To obtain a rapidly linearly converging method ideas from the RPM are used. Write $\Delta x$ as $\Delta x = V\Delta\hat{p} + \Delta q$, and note that $\Delta q = Q\Delta q$ where $Q$ is the orthogonal projector on the complement of span$\{V\}$, $Q = I - VV^T$. Herewith, the correction equation is rewritten as

$$\begin{bmatrix} Q(M-I)Q & Q(M-I)V & Q\varphi_T \\ V^T(M-I)Q & V^T(M-I)V & V^T\varphi_T \\ s_xQ & s_xV & s_T \end{bmatrix} \begin{bmatrix} \Delta q \\ \Delta\hat{p} \\ \Delta T \end{bmatrix} = - \begin{bmatrix} Qr \\ V^Tr \\ s \end{bmatrix}.$$

If $V$ is an invariant subspace of $M$ then $Q(M-I)V = 0$, we will assume that this is the case. Moreover, we will assume that $Q\varphi_T$ is negligible; for the periodic solution $\varphi_T$ is an eigenvector of $M$ with eigenvalue 1 (see the remark above) and hence a vector of the subspace $\mathcal{P}$, therefore $Q\varphi_T$ will be small during the Newton process. This results in a Gauss-Seidel like iteration. First we have to solve for $\Delta q$ and then the other unknowns follow from the reduced system

$$\begin{bmatrix} V^T(M-I)V & V^T\varphi_T \\ s_xV & s_T \end{bmatrix} \begin{bmatrix} \Delta\hat{p} \\ \Delta T \end{bmatrix} = - \begin{bmatrix} V^Tr + V^TM\Delta q \\ s + s_x\Delta q \end{bmatrix}. \tag{6.7}$$

In order to solve for $\Delta q$ we need to solve a system with the matrix $Q(M-I)Q$. This forms the bottle-neck in the computations since we do not want to compute this full matrix explicitly. However, we can compute a multiplication of $M$ with a vector by numerical differentiation as in (6.4). Hence we can try to approximate the inverse of $Q(M-I)Q$ by a polynomial. In the present version a truncated Neumann series is used

$$[Q(M-I)Q]^+ \approx Q + QMQ + (QMQ)^2 + ... + (QMQ)^k, \tag{6.8}$$

where $+$ denotes the pseudo-inverse; this yields exactly the inverse of $M-I$ restricted to the orthogonal complement of $V$. The value of $k$ is specified by the user (in our applications we have used $k=10$ and $k=20$).

Once the reduced system is computed, its solution can be found by a direct method since the system is small (in our case of order 30).

*Automatic phase condition through pseudo-inverse* In order to avoid giving a phase condition a variant based on least-squares is also available. This variant has been used in our computations. Here, the reduced system to be solved is

$$\left[\begin{array}{cc} V^T(M-I)V & V^T\varphi_T \end{array}\right] \left[\begin{array}{c} \Delta\hat{p} \\ \Delta T \end{array}\right] = -\left[\begin{array}{c} V^T r + V^T M \Delta q \end{array}\right]. \tag{6.9}$$

This system is underdetermined. Hence, the solution is not unique. A way to make it unique is to find the solution with minimal two-norm. This solution is found by employing the pseudo-inverse, which implicitly defines a phase condition. We will make the form of this condition more explicit in the following.

In case of a periodic solution $\varphi_T$ is the eigenvector of $M$ with eigenvalue 1. In general this is a simple eigenvalue and due to the construction in the Newton-Picard process it is contained in the invariant subspace $V$. Hence, if $[\,\overline{\Delta\hat{p}}\,,\,\overline{\Delta T}\,]^T$ is a solution of the underdetermined system then $[\,\overline{\Delta\hat{p}}+\mu V^T\varphi_T, \overline{\Delta T}\,]^T$ is a solution as well. Minimization of the length of this vector leads to the condition that the inner product $(V^T\varphi_T, \overline{\Delta\hat{p}}+\mu V^T\varphi_T) = 0$. Or in other words, the unique solution of the system has to be perpendicular to $[V^T\varphi_T, 0]^T$. This is the implicit phase condition when the system is solved using the pseudo-inverse. In [31, 32] it is shown that this condition is close to a desirable one.

We have also used the code on the steady branch. In that case, $V^T(M-I)V$ is in general non-singular, but $\varphi_T = 0$ and hence $V^T\varphi_T = 0$. Then the solution with minimal two-norm is the one with $\Delta T = 0$, which is precisely what we want.

As usual in the application of the pseudo-inverse the user has to specify a drop tolerance for the singular values. Here, a singular value is dropped if it is less than the drop tolerance times the maximum singular value.

*Stopping criterion* The stopping criterion of the Newton-Picard process is based on the residual and the corrections. More precise, the root mean square of both the residual, $\Delta p$ and $\Delta q$ and the magnitude of $\Delta T$ should be less than a user specified tolerance.

*Pseudo-arclength method* In continuation often pseudo-arclength parametrization is used in order to be able to pass through a fold point. Then, an equation has to be added to the system. When the pseudo-arclength parameter is denoted by $\mu$, this equation is given by

$$n(x, T, \gamma, \mu) = 0.$$

The precise form is given in formula (2.1) of Lust and Roose. In our computations we did not use this pseudo-arclength variant.

## 6.5 Numerical results

The PDECONT tool can be used to compute both steady and periodic flows and their stability. For the lid-driven cavity problem depending on the Reynolds number both types of flows can occur. In this section we will present results from computations on both a steady and a periodic branch. All computations have been performed on a $128 \times 128$ stretched grid.

### 6.5.1 Along the steady solution branch

As a starting solution for the Newton-Picard process we computed the steady-state solution at Re=7500. This was done by a separate program using the backward Euler method ($\theta = 1$).

#### Setting of the parameters in PDECONT

The tool PDECONT is a cascade of iterative procedures, we will shortly indicate the criteria used for each procedure.

*Continuation process* In the continuation process the Reynolds number is used as the continuation parameter. The process is started at Re=7500 with step size 250. During the process the step size is automatically adjusted. The smallest and largest step size we allow are 25 and 500, respectively.

*Subspace iteration* During the computation we keep the dimension of span$\{V\}$ fixed at 22. For the invariant subspace we require that the stopping criterion is satisfied with tolerance 0.01 for all eigenvalues greater than 0.95. The parameter used in the numerical differentiation is $10^{-5}$. This is small but necessary for two reasons. First of all, we obviously want that the error is small. Secondly, we have to deal with the non-linearity in case of a complex eigenvalue. To be more precise, suppose that a certain eigenvalue is complex then the two eigenvectors span a two-dimensional space. In the algorithm we iterate on two real orthogonal vectors which approximately span this space. Application of the nonlinear matrix $M$ gives as result two new real vectors which are orthogonalized in order to give the new basis. Since $M$ is nonlinear the respective spaces built in this way will not converge when the parameter in the numerical differentiation is too large. The criterion in the time stepping process is also very important in this respect.

*Time integration* We set the time step at 0.125. This is adapted by the program such that its multiple fits in the period. For the Newton method a stopping criterion of $10^{-12}$ is imposed on the correction (maximum norm). This is needed for the numerical differentiation in order to have a sufficient small error in $MV$. Since the periods we are computing are of order 1 about ten time steps are needed yielding an error in the integration over one period of about $10^{-11}$. Due to the numerical differentiation this error will become $10^{-6} (= 10^{-11}/10^{-5})$. We experimented with this stopping criterion and observed that this high accuracy was needed to maintain convergence of the eigenvalues. We think that also the non-normality of $M$ will play an important role here.

The problem to find the magnitude of variation in the numerical differentiation is absent when using the differential equation for the perturbation, using the exact Jacobian. We are inclined to do this in future experiments. We need only half of the accuracy for the time integration. So it is equally expensive.

*Solving correction equation* The value of $k$ in the Neumann series (6.8) in the Picard step is set to 10. For the computation of the pseudo-inverse of the reduced system (6.9)

the drop tolerance for the singular values is $10^{-4}$. The tolerance for the stopping criterion of the Newton-Picard process is 0.005.

### Computation of eigenvalue data

In Table 6.2 eigenvalues and corresponding periods and frequencies are given. In each column one can see how an eigenvalue evolves as a function of the Reynolds number. A plot of this data is given in Figure 6.2. These eigenvalues are computed from equation (6.1) as discussed in Section 6.3.2. Hence, the eigenvalues are independent of $\theta$. The results indicated with † in the table are not fully converged. These results usually belong to an eigenvalue with a large imaginary part for which the amplification factor (see Section 6.3.2) is not in the range of accurately computed amplification factors.

| Re | $\lambda_1$ | $T_{est}$ | $1/T_{est}$ | $\lambda_2$ | $T_{est}$ | $1/T_{est}$ |
|---|---|---|---|---|---|---|
| 7500 | -0.0159± 0.9406i | 6.6799 | 0.1497 | | | |
| 7750 | -0.0152± 0.9398i | 6.6858 | 0.1496 | -0.0163± 2.7813i | 2.2590 | 0.4427 |
| 8000 | -0.0144± 0.9392i | 6.6897 | 0.1495 | -0.0068± 2.7762i | 2.2633 | 0.4418 |
| 8375 | -0.0138± 0.9371i | 6.7049 | 0.1491 | 0.0009± 2.7640i | 2.2732 | 0.4399 |
| 8875 | -0.0130± 0.9357i | 6.7152 | 0.1489 | 0.0189± 2.7518i | 2.2833 | 0.4380 |
| 9375 | -0.0124± 0.9345i | 6.7233 | 0.1487 | 0.0313± 2.7354i | 2.2970 | 0.4353 |
| 9875 | -0.0115± 0.9322i | 6.7403 | 0.1484 | 0.0473± 2.7247i | 2.3060 | 0.4337 |
| 10375 | -0.0110± 0.9330i | 6.7346 | 0.1485 | 0.0608± 2.7148i | 2.3144 | 0.4321 |
| Re | $\lambda_3$ | $T_{est}$ | $1/T_{est}$ | $\lambda_4$ | $T_{est}$ | $1/T_{est}$ |
| 7750 | -0.0245± 1.8672i | 3.3650 | 0.2972 | -0.0391± 0.9355i | 6.7166 | 0.1489 |
| 8000 | -0.0268± 1.8667i | 3.3660 | 0.2971 | -0.0375± 0.9347i | 6.7222 | 0.1488 |
| 8375 | -0.0241± 1.8632i | 3.3723 | 0.2965 | -0.0360± 0.9335i | 6.7310 | 0.1486 |
| 8875 | -0.0223± 1.8598i | 3.3784 | 0.2960 | -0.0336± 0.9312i | 6.7471 | 0.1482 |
| 9375 | -0.0220± 1.8571i | 3.3833 | 0.2956 | -0.0314± 0.9301i | 6.7556 | 0.1480 |
| 9875 | -0.0214± 1.8545i | 3.3881 | 0.2952 | -0.0296± 0.9289i | 6.7641 | 0.1478 |
| 10375 | -0.0194± 1.8489i | 3.3983 | 0.2943 | -0.0274± 0.9264i | 6.7822 | 0.1474 |
| Re | $\lambda_5$ | $T_{est}$ | $1/T_{est}$ | $\lambda_6$ | $T_{est}$ | $1/T_{est}$ |
| 8375 | -0.0312± 3.7956i† | 1.6554 | 0.6041 | | | |
| 8875 | 0.0275± 3.8058i | 1.6509 | 0.6057 | -0.0151± 3.3252i† | 1.8896 | 0.5292 |
| 9375 | 0.0583± 3.7903i | 1.6577 | 0.6032 | 0.0259± 3.2788i | 1.9163 | 0.5218 |
| 9875 | 0.0871± 3.7818i | 1.6614 | 0.6019 | 0.0490± 3.2702i | 1.9214 | 0.5205 |
| 10375 | 0.1136± 3.7755i | 1.6642 | 0.6009 | 0.0717± 3.2565i | 1.9294 | 0.5183 |
| Re | $\lambda_7$ | $T_{est}$ | $1/T_{est}$ | $\lambda_8$ | $T_{est}$ | $1/T_{est}$ |
| 8875 | -0.0348± 4.4752i† | 1.4040 | 0.7123 | | | |
| 9375 | 0.0303± 4.3597i | 1.4412 | 0.6939 | | | |
| 9875 | 0.0640± 4.3418i | 1.4471 | 0.6910 | -0.0120± 4.8817i‡ | 1.2871 | 0.7769 |
| 10375 | 0.0952± 4.3249i | 1.4528 | 0.6883 | 0.0291± 4.8370i | 1.2990 | 0.7698 |

Table 6.2: Eigenvalues and corresponding estimated periods and frequencies.

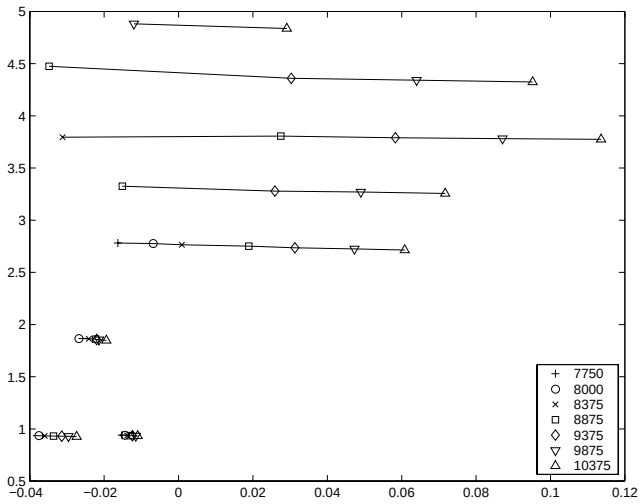


Figure 6.2: Eigenvalues in the complex plane for various Reynolds numbers.

We verified the results by requiring two extra digits of accuracy for the solution at Re=9875. It appeared that the maximal difference in the eigenvalues was less than 14e-4. Hence the above results are rather accurate. We also halved the time step and decreased $\theta$ to 0.505. Now, we observed the result indicated with a ‡ in the table. This component is unstable for Re=10,375 as we can see from the last result in the table. Such a component with a large imaginary part is indeed not seen if $\theta$ is too large.

### Comparison to Cazemier's results

From Table 6.2 we see that in our computations the first bifurcation occurs in the neighbourhood of Re=8375. The difference with Cazemier's DNS value 7972 is due to our coarser grid and lower-order discretization.

In Cazemier's analysis with the low-dimensional dynamical system the frequency 0.60 Hz appeared at the first transition point, whereas with both Cazemier's DNS and our analysis the frequency 0.44 Hz occurred. In our results the frequency 0.60 Hz seems to be connected to the eigenvalue which causes the second Hopf bifurcation and has the largest real part for the higher Reynolds numbers. Hence, it does not surprise us that this frequency plays a prominent role.

Comparing our results for Reynolds number 10,000 to those of Table 6.1 we observe many common components, the remaining frequencies in that table can almost always be formed by a combination of the frequencies that we have found. To be more precise, from Table 6.2 we find for Re=10,000 the unstable modes with frequencies 0.43, 0.52, 0.60, 0.69 Hz which correspond to the frequencies 0.44, 0.53, 0.60, 0,70 Hz in Table 6.1. These are the frequencies of the Hopf bifurcations, hence the others must be integer combinations of them. In Table 6.3 we give a possible realization of the remaining frequencies in the same order as in Table 6.1. The most important frequency in Table 6.1 is 0.27 Hz, which has a large amplitude in Cazemier's DNS computations and which was believed to originate from a Hopf bifurcation. From our computations we find that it is a combination frequency.

This is confirmed by the computation along the periodic solution branch starting from the period corresponding to the frequency 0.44 Hz, see Section 6.5.2. The first unstable mode on this branch is the one corresponding to the frequency 0.70 Hz.

| $f$ | reconstr. |
|---|---|
| 0.27 | 0.70-0.44 |
| 0.44 | Hopf |
| 0.11 | 0.70-0.60 |
| 0.16 | 0.60-0.44 or 0.70-0.53 |
| 0.32 | 2(0.60-0.44) |
| 0.53 | Hopf |
| 0.70 | Hopf |
| 0.06 | 0.60-0.53 |
| 0.38 | (0.70-0.44)+(0.70-0.60) |
| 0.60 | Hopf |
| 0.88 | 2(0.44) |

Table 6.3: Reconstruction attempt of unstable modes.

### 6.5.2 Along periodic solution branches

As mentioned before the first bifurcation in our computations occurs in the neighbourhood of Re=8375. This is determined by computing the eigenvalues of the Jacobian with equation (6.1), and looking when they pass the imaginary axis. However, if we want to switch to the periodic solution branch, we have to consider the transition of the time-integration process, i.e. to determine where the amplifier $r$ leaves the unit circle. This may shift the transition point, because the eigenvalues do not depend on the time-integration method whereas the amplification factor $r$ does.

We increased the accuracy of the time integration by using $\theta = 0.502$ and $\Delta t = 0.06125$. The continuation process was started again at Re=8000 with step size 100. We increased the dimension of the invariant subspace to 30 and used $k = 20$ in the truncated Neumann series (6.8). At the start of this extension we see an algebraic effect on the convergence behaviour. After a while the amplification factors of the additional vectors settle down at about 0.78 which complies with equation (6.2) which gives approximately 0.74. This is quite close to our region of interest.

We again observed that the steady solution becomes unstable at Re=8400 for a perturbation with period about 2.3 seconds. We started on the periodic branch at Re=8550 using the steady solution plus 0.5 times the first vector in the basis, i.e. the vector belonging to $\lambda_2$. The factor was found after some trials (0.05 still returned the steady solution). A plot of how the Floquet multipliers move is shown in Figure 6.3. The marks in this plot make it possible to follow the Floquet multipliers for increasing Reynolds numbers; they follow two consecutive sequences of $\circ, \times, +, *, \square, \diamond$. All occurring multipliers can be traced back to the components given in Table 6.2. The radius of the Floquet multipliers is scaled in order to stretch the plot near the unit circle. On the dash-dotted and dashed line the radius is 0.90 and 0.95, respectively.

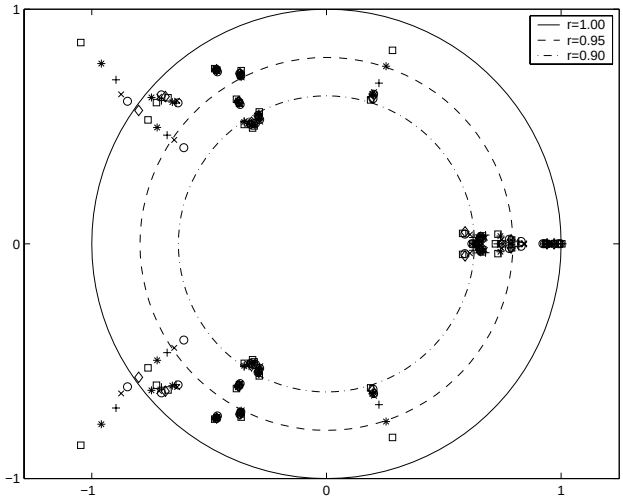


Figure 6.3: Floquet multipliers of the periodic solution emerging at the first Hopf bifurcation for Reynolds numbers in the range 8550-10,000. The radius of the values is scaled with $100^{r-1}$.

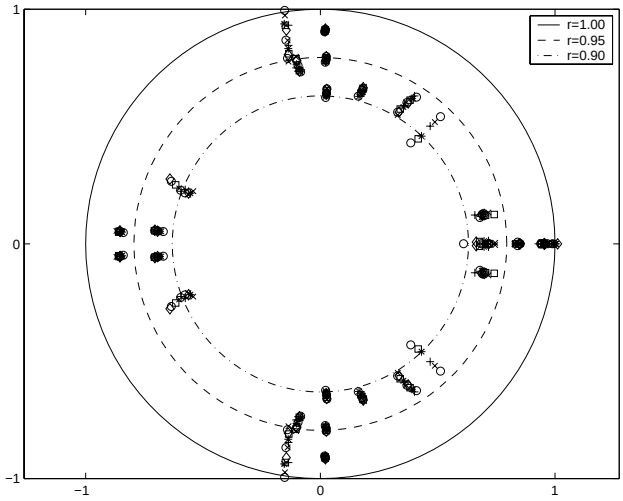


Figure 6.4: Floquet multipliers of the periodic solution emerging at the second Hopf bifurcation for Reynolds numbers in the range 8700-10,000. The radius of the values is scaled with $100^{r-1}$.

From Figure 6.3 we observe that at Re=9150 a pair of complex conjugate Floquet multipliers leaves the unit disc and hence the periodic solution becomes unstable. The frequency of the unstable mode is about 0.70. This is again computed with equation (6.1), which holds approximately now. This component is clearly generated from $\lambda_7$ in Table 6.2. Since a pair of complex conjugate Floquet multipliers has left the unit disc a 2-periodic or quasi-periodic solution will emerge, we can not compute such solutions

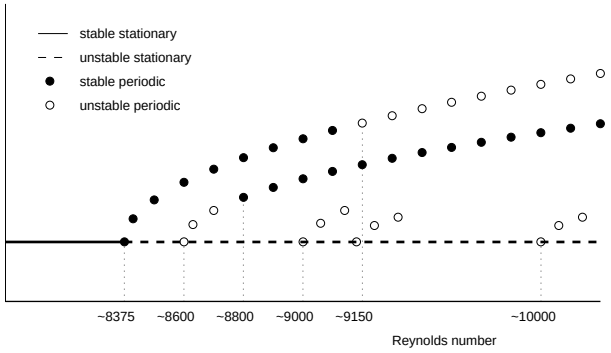


Figure 6.5: Bifurcation diagram.

with our method. From Re=9250 we followed the unstable periodic solution branch with increments 250 until we reached Re=10,000. No new unstable modes occur.

To compute the periodic branch emerging from the second bifurcation point at the steady-state branch, we started the computation at Re=8700 using the steady solution plus 0.5 times the vector corresponding to the eigenvalue $\lambda_5$ which causes the bifurcation. The Reynolds number is increased with step size 100 until Re=10000. The movement of the Floquet multipliers is shown in Figure 6.4.

At this second bifurcation point the steady-state solution remains unstable. The periodic solution has period 1.7 seconds, and from the Floquet multipliers in Figure 6.4 we see that the emerging periodic solution is unstable. However, we see that at about Reynolds number 8800 a pair of complex conjugate Floquet multipliers reenters the unit disc and hence at that Reynolds number the periodic solution becomes stable and a 2-periodic or quasi-periodic solution will disappear. The periodic solution remains stable for the remainder of the computation to Re=10000. Since the DNS results from Cazemier show a stable periodic solution with frequency 0.6 Hz. (period 1.7 seconds) at Re=12000 we think that in between this periodic solution is stable as well.

### 6.5.3 Bifurcation diagram

In Figure 6.5 a partial bifurcation diagram based on our results is shown. This picture only gives a qualitative behaviour of the system. We see that when the stationary solution becomes unstable at Reynolds number 8375 a periodic branch occurs. This periodic branch is stable up to a Reynolds number of about 9150. At Reynolds number of about 8600 a second periodic branch emerges from the steady-state branch. This periodic branch is unstable up to Reynolds number of about 8800, where it becomes stable. We observe that for Reynolds numbers in the range from 8800 to 9150 two stable periodic solutions exist. From the unstable stationary branch other periodic branches occur at Reynolds numbers of about 8600, 9000, 9100 and 10,000. These branches are all (initially) unstable.

## 6.6 Conclusions

We split our conclusions in two parts: the numerical approach and the bifurcation behaviour. With respect to the numerical approach we mention:

- Large scale bifurcation analysis of periodic solutions is possible. However, the current version of PDECONT consumes a lot of computer time, which is due to the slow convergence of the invariant subspace.
- The use of an implicit method is possible with good iterative solvers, such as CG type methods preconditioned by an MRILU factorization. Special care has to be taken that only the Floquet multipliers of interest are found.
- The space discretization is chosen such that it remains stable if no artificial viscosity is used, this is an important reason why we find bifurcations at low Reynolds numbers.

With respect to the bifurcation behaviour on the range Re=7500 to 10,000 we find the following:

- The first Hopf bifurcation occurs with our model at about Re=8375. This is a bit higher than the value of Cazemier Re=7972, which we attribute to the fact that our grid is twice as coarse and only second-order accurate discretizations are used.
- The next Hopf bifurcations occur at about Re=8600, 9000, 9100 and 10,000. The frequencies corresponding to the corresponding modes are also encountered on the branches of periodic solutions. They can also be used to explain the time signals found by Cazemier.
- The periodic solution occurring at Re=8375 is stable to about Re=9150. Thereafter, no new unstable modes occur before Re=10,000.
- The periodic solution emerging at Re=8600 is unstable until about Re=8800. After that point it is stable to at least Re=10,000.